TM No. 861018

NAVAL UNDERWATER SYSTEMS CENTER
NEW LONDON LABORATORY
NEW LONDON, CONNECTICUT 06320

Technical Memorandum

INTERFACE AND DATA ACQUISITION PROGRAMS FOR THE MICROVAX COMPUTER AND D6000 WAVEFORM ANALYZER

Date: 12 March 1986

Prepared by: Walter S. Hauck III
Walter S. Hauck III
Electronics Engineer

Pat Maciejewski
Pat Maciejewski
Computer Scientist

Report Documentation Page

Form Approved OMB No. 0704-0188

1. REPORT DATE
12 MAR 1986

2. REPORT TYPE
Technical Memo

3. DATES COVERED
12-03-1986 to 12-03-1986

4. TITLE AND SUBTITLE
Interface and Data Acquisition Programs for the Microvax Computer and 06000 Waveform Analyzer

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Walter Hauck; Pat Maciejewski

5d. PROJECT NUMBER
A67001 and A65000

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Underwater Systems Center,New London,CT,06320

8. PERFORMING ORGANIZATION REPORT NUMBER
TM No. 861018

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES
NUWC2015

14. ABSTRACT
This memorandum describes a set of FORTRAN programs used to interface a 06000 Universal Waveform Analyzer and a MicroVAX I computer. Required program inputs and the physical connections between analyzer and computer are discussed in detail.

15. SUBJECT TERMS
FORTRAN; High Frequency Acoustics; EVA Support for Shipboard Sonar; MicroVAX; Universal Waveform Analyzer

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			
unclassified	**unclassified**	**unclassified**	**Same as Report (SAR)**	**18**	

TM No. 861018

NAVAL UNDERWATER SYSTEMS CENTER
NEW LONDON LABORATORY
NEW LONDON, CONNECTICUT 06320

Technical Memorandum

INTERFACE AND DATA ACQUISITION PROGRAMS FOR THE MICROVAX COMPUTER AND D6000 WAVEFORM ANALYZER

Date: 12 March 1986

Prepared by: Walter S. Hauck III
Walter S. Hauck III
Electronics Engineer

Pat Maciejewski
Pat Maciejewski
Computer Scientist

PREFACE

This work was supported under NUSC Project No. A67001, High Frequency Acoustics, Principal Investigator, Dr. William Roderick, and A65000, EVA Support for Shipboard Sonar, Principal Investigator, Peter Herstein. The sponsoring activities are the Naval Ocean Research and Development Activity, Program Manager, Dr. Robert Farwell, and the Naval Research Laboratory, L. B. Palmer, EVA Block Manager.

The authors would like to thank D. Granz, of Apollo Computer, for the original D6000 interface software, J. Nordquist, Code 3332, for programming support and suggestions, and G. Botseas, Code 3332, for his help and suggestions throughout the course of his work.

ABSTRACT

This memorandum describes a set of FORTRAN programs used to interface a D6000 Universal Waveform Analyzer and a MicroVAX I computer. Required program inputs and the physical connections between analyzer and computer are discussed in detail.

INTRODUCTION

The purpose of this memorandum is to provide a guide for interfacing a MicroVAX I computer with an Analogic D6000 Universal Waveform Analyzer, and to provide a reference document for the interface and data acquisition software developed to support this data acquisition system.

This memorandum is divided into three sections; 1) a description of the MicroVAX I - D6000 data acquisition system, 2) description of how to use VAX programs to acquire data with the D6000, and 3) transfer of data from the D6000 to VAX. It is assumed throughout the memorandum that the reader is familiar with the VMS operating system and operation of the D6000 waveform analyzer.

I. DATA ACQUISITION SYSTEM DESCRIPTION

1. Description

The system described in this memorandum was developed as a sea-going data acquisition system. It was successfully taken aboard ship during the High Frequency Acoustics 85 Experiment. Figure 1 is a diagram of the physical connections between the D6000 and MicroVAX, as well as the acquisition program directory architecture. The D6000 can acquire data simultaneously on four channels, with an aggregate sampling rate of 100 KHz. In addition, an external trigger and clock frequency may be input. The user is referred to the D6000 manual for a complete description. The physical connection between D6000 and MicroVAX is a standard null modem cable, pins 2 and 3 reversed, connected to the SER1 port on the D6000 and a terminal port, TTA1 in this case, on the MicroVAX. Alternatively, a standard modem cable may be connected to the SER0 port on the D6000. Figure 2 shows the location and designation of the serial ports on the D6000 backplane.

2. Configuring the D6000 SER1 Port

Figure 3 is a diagram of the front of the D6000 analyzer. The three basic type of keys are data acquisition, black on the front panel, keypad keys, and softkeys, black and located beneath the display. To change a communications setting, either the data acquisition key, such as IO, or a combination of data acquisition keys must be pressed. This brings up different fields on the display above the softkeys, which are pressed to modify parameters. The keystrokes required to set up the SER1 port for VAX communication are;

1) IO (data acquisition key labeled IO) Screen displays

```
>I/O Field Format
    Line Format
    Message Format
    GPIB Configuration
    RS-232 Configuration
```

2) Press 2nd softkey from left 5 times until arrow is next to RS-232 Configuration label

3) Press I/O key again. Softkey Fields displayed are now;

PORT	FLD LEN	FLD DLM	FORMAT
GPIB	10	COMMA	SCI

4) Hit the left-most softkey until SER1 appears beneath PORT

5) Press the DIR and PROG keys together. Both key lights should be on. Softkey fields are now;

CMD DEV	LOG DEV	ERROR MODE	CONTROL
GPIB	<NONE>	LOGGED	LOCAL

6) Press left most softkey until SER1: appears under CMD DEV

7) Press third set of softkeys until IMMEDIATE appears under ERROR MODE

SER1 is now ready for VAX communications.

3. Configuring MicroVAX port

The port connected to the D6000 must first be made available to all users with the command

```
$ SET PROT=(W:RWLP)/DEVICE TTA1:
```

The necessary terminal characteristics are set with the command

```
$ SET TERM/PERM/PASS/NOTTYSYNC/FULLDUP-
$NOHOSTSYNC/DEVICE=UNKNOWN/SPEED=9600 TTA1:
```

The terminal port should now be ready for data transfer between the VAX and D6000.

4. Using Other VAXes

The D6000 has also been successfully interfaced with a VAX 11/780 computer, NUSCNET node name V331. Figure 4 is a diagram of the physical connections and program directory structure. This system will function in an identical manner to the MicroVAX I system diagrammed in Figure 1. The descriptions of data acquisition and transfer programs which follows is also

applicable for a D6000-VAX 11/780 combination. The commands described in this memorandum should apply to either VAX system, with any differences transparent to the user.

II. D6000 CONTROL PROGRAMS FOR DATA ACQUISITION

Several programs are available for controlling data acquisition on the D6000. A summary of each program is provided below.

TALK - Interactive I/O with D6000.

TALK is the basic I/O program between the MicroVAX and D6000. To begin TALK, issue the command

```
$ RUN [D6K]TALK
```

the program returns with

```
INPUT D6000 COMMAND
```

Enter any valid D6000 command, in capital letters and hit return. The program echos back the command and the D6000 response if any. END TALK by typing EXIT followed by a carriage return.

TALK calls subroutines stored in the D6K library.

TALK2 - Runstream version of TALK.

TALK2 is a version of TALK that accepts inputs from logical unit 11, instead of unit 5. TALK2 is used in all data transfer programs described in the next section.

TALK calls subroutines stored in the D6K library.

2TALK - TALK for controlling 2 D6000s.

Figure 5 is a connection diagram for controlling 2 D6000s from the MicroVAX. The terminal ports are TTA0 and TTA1. 2TALK sends successive commands to different D6000s, i.e., the first command is sent to the D6000 connected to TTA1, the second command to TTA0, the third to TTA1, etc. Like TALK, 2Talk is ended with the command EXIT.

CTL - Multiple named files with error file checking.

CTL is the D6000 controller program for acquiring a series of sequentially named files on the D6000. It is designed for use when an external trigger is used to capture a short waveform. To insure that a series of events are captured without any loss of data, CTL continuously checks the trigger status. A trigger is declared if the D6000 trigger state, controlled by the D6000 mnemonic QTRG, changes from 2 to 3 or 4. If the trigger is in transition when the query is made, the trigger state is unreadable, and an error message is sent by the D6000. In this case, CTL causes the D6000 to

keep the file, with a file named preceded by an "E", for error file. The data transfer program GET compares an error file with the valid data files preceding and following it, and deletes the error file if it is identical to a valid data file. If the error file contains different values than the surrounding files, it is assumed a valid trigger occurred, and the error file is renamed.

The CTL prompts are:

```
INPUT NUMBER OF FILES TO BE NAMED:
INPUT NUMBER OF CHANNELS IN USE:
INPUT CHANNEL NUMBERS AND TIME BASE "A" OR "B":
     EXAMPLE "A1" FOR TMB A, CHAN 1
INPUT FILE NAME FOR CHAN XX
          .
          .
          .
     repeats for each channel
PRESS RETURN WHEN READY TO ACQUIRE DATA
```

File names are restricted to three characters, or combinations of characters and numbers. Any combination of channels and timebases may be used. There are no restrictions on the number of files that may be acquired by CTL, except the 50000 maximum sample storage of the D6000 memory. Note: The total number of files collected with CTL is the sum of the valid data files plus the error file, so that the actual number of valid data files acquired may be less than the number of files requested.

CTL uses subroutines from the D6K and DNAMES libraries.

CTLA - CTL for 2 D6000s.

CTLA is a version of CTL for acquiring data on 2 D6000s. The first trigger on the D6000 attached to TTA1 causes that analyzer to acquire a data file. This analyzer will ignore all triggers until a valid trigger is received on the D6000 attached to TTA0. If the trigger status is unreadable, an error file is acquired and control passed back to the other D6000. Prompts are similar to CTL, with the setup on the D6000 on TTA1 requested first, and TTA0, second. The channel numbers and time bases need not be identical between analyzers.

CTLA uses subroutines from the D6K and DNAMES libraries.

2CTL - CTL for 2 D6000s without error files.

2CTL is a version of CTLA, described above, with the program logic modified for handling unreadable trigger status. If the trigger status is unreadable, the analyzer waits until it receives a valid trigger. Data is collected, and control passed to second D6000. acquired data on an analyzer is not necessarily from sequential triggers.

2CTL uses subroutines from the D6K and DNAMES libraries.

D6KTERM - Setting the VAX terminal port for D6000 communications.

If the D6000 is attached to a terminal port not configured for the D6000 communications software, the command procedure D6KTERM may be used. The command

$ @D6KTERM TTA3:

will configure the MicroVAX port TTA3 for the D6000, as described in Section I.3. The user must have the system privilege to allocate the desired port for D6KTERM to function properly.

D6KSET - Setting up the D6000 from a command file.

After the D6000/MicroVAX link has been established, i.e. the TALK program functioning, commonly used D6000 set up parameters may be sent to the D6000 using the command procedure D6KSET. The D6000 commands are store in a file, with the name [D6K]*.D6K, where * is the file name. The set up file is created using any VAX editor. The commands should be in upper case letters and start in column 1.

III. DATA TRANSFER FROM D6000 TO MICROVAX

Data files are transferred between the D6000 and MicroVAX in two steps. The first step transfers the data file from the D6000 into a temporary file of the same name in the directory [TEMP]. These files are a combination of ASCII characters and numbers, and are unreadable as data files. The format of these files is then "fixed" using a second set of programs, and placed in the data directory [DATA]. The temporary file are deleted when the re-formatted files are created. The programs described in this section have one of two functions, either to get data, or to fix data.

GET - Transfer D6000 files and check for error files.

GET is the data transfer command procedure for sets of files created on the D6000 with the program CTL. GET prompts the user for;

FILE NAME PREFIX:
STARTING FILE NUMBER:
TOTAL NUMBER OF FILES TO BE PROCESSED:

where the file name prefix is the three character name used with CTL. The user may start at any file number, and transfer any number of sequentially numbered files. GET also attempts to find any error files created by CTL. The resulting data transfer with GET is the slowest of the transfer programs. When complete, the directory [TEMP] contains all the requested files and companion error files.

GET uses TALK2 to do the actual file transfer, and calls the command procedure MOVE.

FIX - Reformat program for GET.

The files transferred with GET are reformatted using the command procedure FIX. FIX reformats each data file using the program FIXIT and checks to see if the length of the companion error file is greater than 50 characters. If it is, the error file is also reformatted, and the numbers in the error file compared to the numbers in the data file with the same file name and number. If the numbers are different, FIX assumes the error file is valid data and renames the file, removing the "E" prefix. If the numbers in the error file and data file are identical, the error file is deleted, and FIX continues with the next data file. If the length of the error file is less than 50 characters, FIX deletes the file and continues. After reformatting the data, FIX copies the remaining files into the [DATA] directory, and deletes the versions left in [TEMP]. The program prompts are identical to GET.

Note: it is not necessary to call FIX immediately after transferring the data files with GET. When it is important to acquire the data on the D6000, transfer the data, and re-acquire more data as rapidly as possible, FIX may be run with several sets of files in [TEMP]. However, FIX performs a number of checks to insure the data transfer was performed properly, that are not possible during GET. Thus, if FIX runs correctly, the data files are copies of the data acquired by the D6000. If during the data transfer, an error occurred, and the D6000 data files deleted for new data, it is generally not possible to recover the corrupted data files.

FIX calls the programs FIXIT and COMPARE, and the command procedures CLEAN2, SWITCH, TRANSFER and BULK.

PGET - File transfer without error files

PGET is the file transfer command procedure for files created on the D6000 using 2CTL or for transferring CTL files without the error files. The file transfer is roughly twice as fast as GET, since no error files are required. The program prompts are the same as those for GET.

PFIX - Reformatting PGET files

PFIX is a version of FIX that does not use error files. Each data file is reformatted using the program FIXIT and transferred to [DATA]. PGET also deletes any remaining files in [TEMP].

CGET and CFIX - Versions for TTAO

These programs are identical in function to PGET and CGET, but assume the data is transferred from a D6000 connected to terminal port TTAO (see Figure 5).

LONG - Large contiguous data file acquisition.

The alternative to event trigger data acquisition is large record, greater than 10,000 points, acquisition. The intent here is to acquire the maximum amount of data the D6000 can hold, roughly 40,000 data samples. To accomplish this task the D6000 allows the number of points acquired by the time base to be independent from the number of points stored in a channel record. In normal operation, the channel buffer is the same length as the timebase. By decreasing the number of points stored in the channel buffer,

the number of timebase points may be increased beyond 20,000. These points must be copied into the channel buffer in sections, by adjusting the channel buffer offset parameter, BUFOFF.

As an example, assume it is desired to obtain a 30,000 sample record on channel 1, timebase A. The following steps and commands are necessary;

```
REC(1,2)=OFF        (turns off channel 2, timebase A)
REC(1,3)=OFF        (turns off channel 3)
REC(1,4)=OFF        (turns off channel 4)
BUFLEN(1,1)=1000    (sets buffer length to 1000)
BUFOFF=0            (sets the buffer offset to zero)
NPTS=30000          (sets the timebase to 30000 samples)
```

When the trigger level is met, the D6000 will acquire 30000 samples on channels 1, but display only 1000 in buffer BUF.A1. The remainder of the data may be view by changing the buffer offset, BUFFOFF, as shown in Figure 6. Note the data will wrap around for buffer offsets greater than the number of points acquired by the timebase. Data acquired using LONG may be reformatted using the program DIVAFIX.

LONG prompts are;

```
FRAME LENGTH:
BUFFER LENGTH:
OUTPUT FILE NAME:
TIME BASE:
CHANNEL NUMBER:
INITIAL OFFSET:
FINAL OFFSET:
```

LONG creates N files in [TEMP], where N is the buffer length divided by the frame length.

DIVAFIX - Transfer and Reformatting of LONG files

DIVAFIX is the program for transferring and reformatting data collected with the command procedure LONG. The number of points in the channel buffer should be an integral multiple of the number of points in the timebase. If not, DIVAFIX truncates the data at the next smallest integer of the channel buffer length. Each data file is reformatted and placed in [DATA]. In addition, all the data files are concatenated with the same file name as the data file prefix, e.g. the concatenated file created by TEST01 to TEST30 is TEST. The DIVAFIX prompts are identical to GET and FIX.

IV. CONCLUSIONS

The data acquisition system described in this memorandum was successfully used during the High Frequency Acoustics '85 sea test, providing a quick-look analog-to-digital conversion and data analysis system aboard ship. The software discussed in this document, along with the programs described in NUSC

TM No. 861010 provide a thorough, non-real time, data acquisition and analysis capability.

The principal strengths of this system are 1) relative ease of operation, 2) reliability, 3) compatibility with existing VAX software, and 4) low cost, when compared to other systems with similar capabilities.

The drawbacks of this particular data acquisition system are 1) hardware limitations, e.g., maximum 4 data channels and 100 KHz aggregate sampling rate and 2) relatively slow transfer of data between D6000 analyzer and MicroVAX via RS-232 communications. During the High Frequency Acoustics sea test, the data transfer rate proved to be a severe system limitation. Due to the test requirements, the majority of the acoustic data was first recorded on analog tape and later digitized, rather than digitizing all data on the D6000.

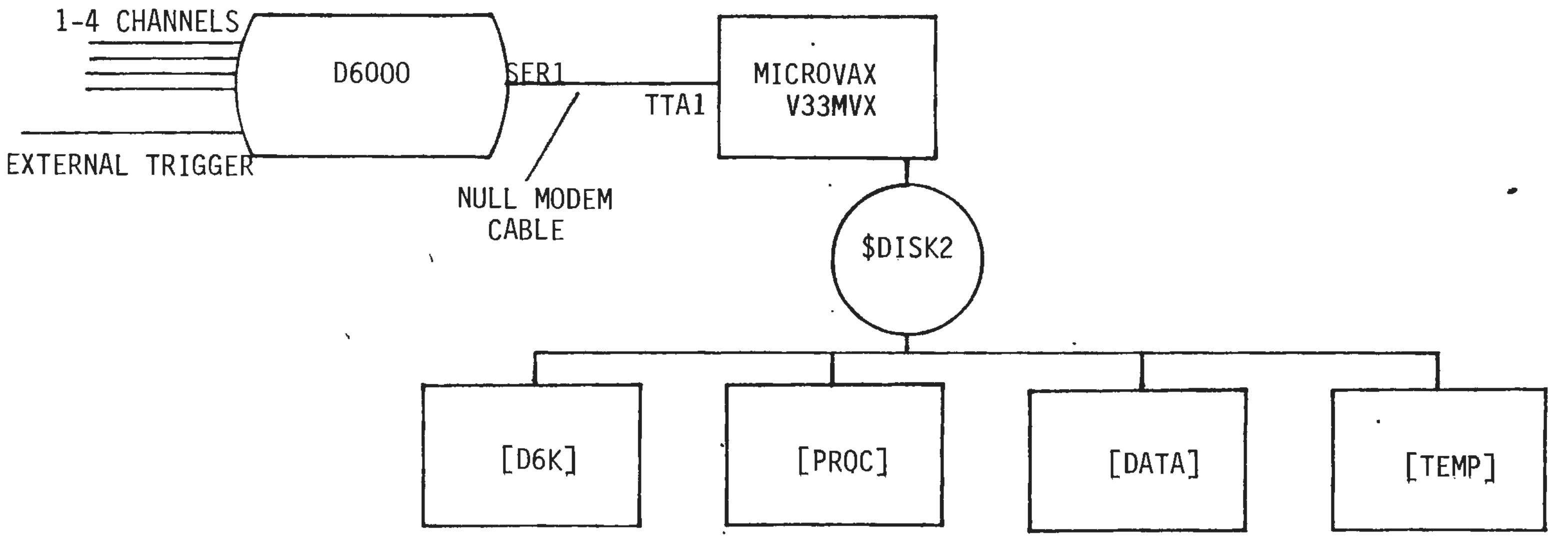

FIGURE 1 - D6000/MicroVAX Connections and Directory Structure.

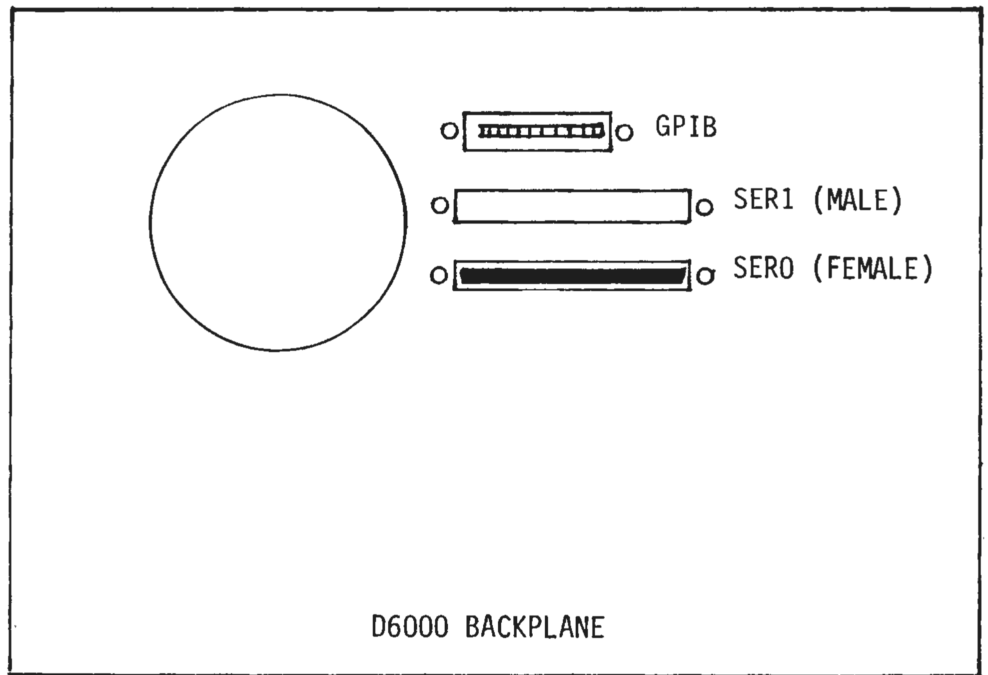

FIGURE 2 - Location of RS-232 Connectors on D6000 Backplane.

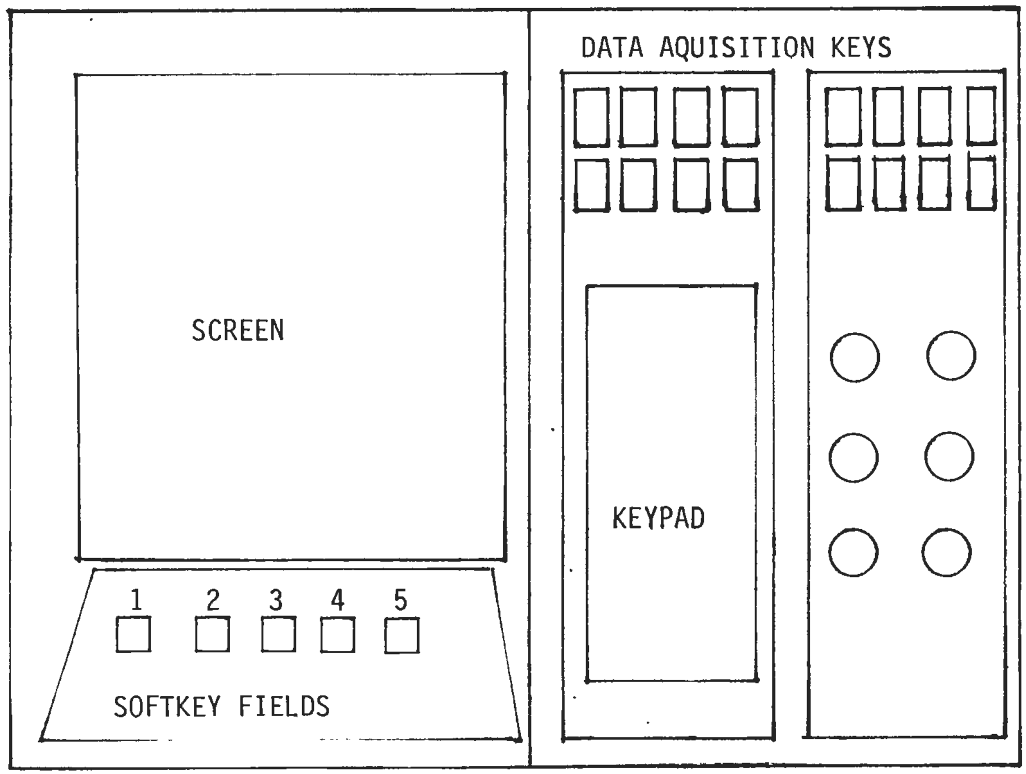

FIGURE 3 - D6000 Front Panel Showing Data Aquistion and Softkey Locations.

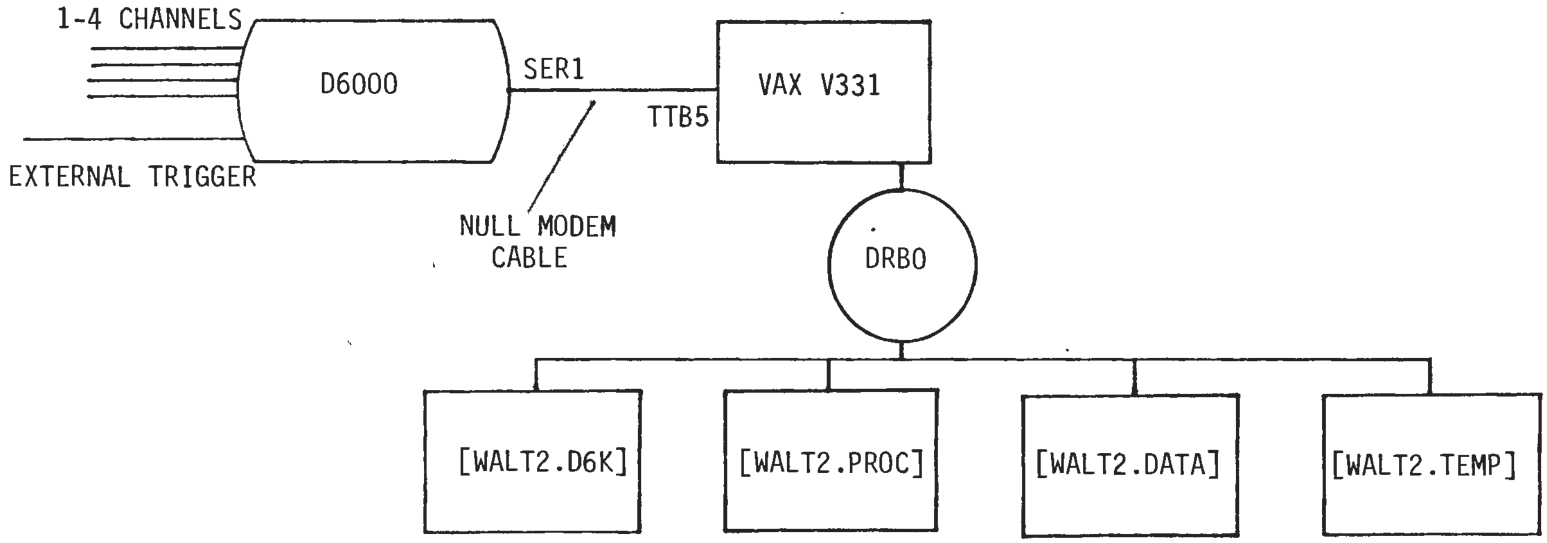

FIGURE 4 - D6000/VAX 11/780 Connections and Directory Structure.

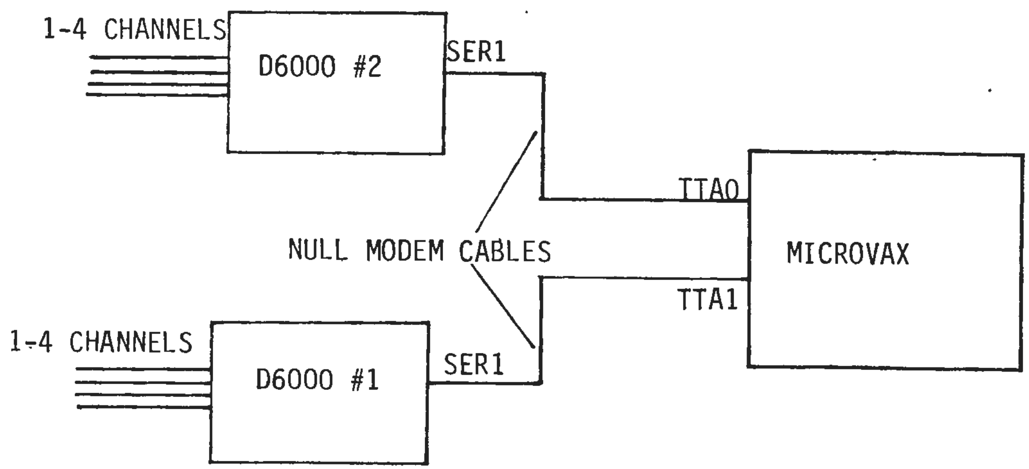

FIGURE 5 - Two D6000s Controlled from MicroVAX.

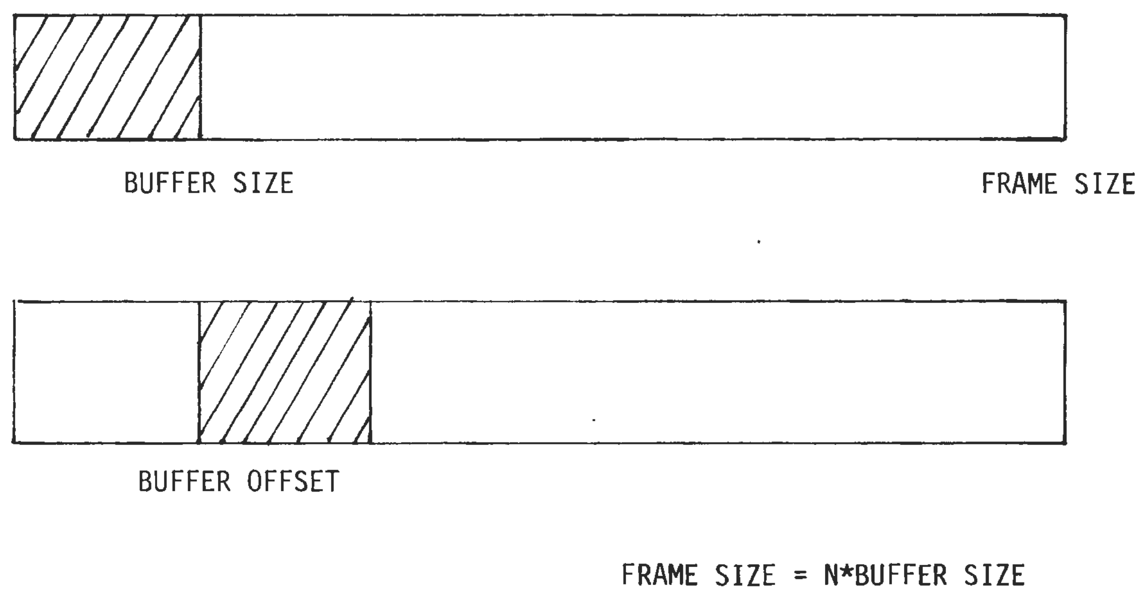

FIGURE 6 - Viewing Small Buffers in a Long Frame by Changing the Buffer Offset.

TM No. 861018

DISTRIBUTION LIST

INTERNAL

Codes:	10	W. Von Winkle
	33	L. Freeman, A. Nuttall, B. Cole
	33A	S. Santaniello, W. Roderick, R. Tompkins
	33A3	P. Herstein, M. Hallisey
	33C	R. Laplante
	331	R. Boivin
	3314	G. Carter, I. Cohen, D. Sheldon, K. Briotti
	332	R. Radlinski
	3322	R. Charette
	3323	D. Bostian, L. Maiocco
	333	J. Hanrahan (Acting)
	333s	
	3331	D. Browning, J. Chester, R. Dullea, P. Koenigs, B. Fisch, M. Fecher, J. Gorman, J. Tattersall, J. Monti, R. Christian, G. Sharman, P. Maciejewski, W. Hauck (5)
	3332	M. Goldstein, R. Deavenport, H. Sternberg, E. Jensen, G. Botseas, D. Potter, J. Nordquist, L. Petitpas
	3333	S. Tate
	70	R. Cote